TYPICALITY FILTERING OF EVENT INDICATORS FOR INFORMATION TECHNOLOGY RESOURCES

FIELD OF THE INVENTION

{001} The invention relates generally to the field of monitoring information technology resources, and more particularly to the field of filtering event indicators such as alerts.

BACKGROUND

{002} As computer equipment has become increasingly complex, the difficulty in monitoring this equipment to keep it functioning properly has become formidable. For example, a server attached to a network should be monitored to ensure that the hardware and software do not malfunction, to ensure that adequate resources such as memory and disk space are available during peak use times, to protect the server from electronic vandalism such as hacking that arrives over the network, and so forth.

{003} As shown in FIG. 1, a monitored device 100 such as a server may be watched over by a monitor 110. The monitor 110 may sense traffic received and transmitted over a network 120 as well as sense conditions both internal and peripheral to the monitored device 100 such as memory use and disk occupancy, CPU utilization, power supply state, cabinet temperature, and numerous other measures of health.

{004} To accomplish these purposes, the monitor 110 typically includes various sensors 111. Here, the term "sensor" is not confined to simple hardware devices, nor is it necessary that the sensors reside literally within the monitor 110. Rather, the term is

intended to encompass both software and hardware systems for sensing the state of parameters that have importance with regard to the proper operation of the monitored device 110. Thus, in correspondence with the examples just mentioned, the monitor 110 may include a sensor that is an intrusion detection system that works in conjunction with protective equipment 130 such as a firewall, another for sensing memory use, yet another for sensing disk occupancy, and so forth. Typically, each sensor determines the state of associated parameters, which state is then evaluated.

{005} Evaluation may involve the use of persistence filters 112. In a simple case, a persistence filter may compare the state determined by the corresponding sensor with a preestablished threshold. If the state violates the threshold, the filter generates an event indicator such as an alert. For example, if the cabinet temperature exceeds ninety degrees Celsius, an alert may be generated. In other cases, the decision process may be more complex as to whether an event indicator should be generated, and if so, what the nature of the indicator should be. For example, a critical alert might be generated if the remaining disk space is determined to be less than 10 MB, a warning alert generated if the remaining disk space is between 10 MB and 25 MB, and an informational alert generated if more than 25 MB remains.

{006} The resulting event indicators may be sent to an event console 140, which may be operated by a human operator 150, or which may be autonomic. The event console 140 or the human operator 150 determine appropriate actions to invoke in response to the event indicators.

{007} Although the configuration of FIG. 1 is now widely used, it suffers from a significant disadvantage: the operating point of the persistence filters 112 must be set as

a compromise that minimizes the generation of both false positives and false negatives. Here, a false positive occurs when an event indicator is generated in response to an unimportant state. Typically, the human operator 150 exercises independent judgment and may decline to invoke a response to a false positive. On the other hand, a false negative occurs when an event is not generated despite the existence of a critical state that requires attention.

{008} Because false negatives are generally more damaging than false positives, there is a tendency to configure the persistence filters to err on the side of permissiveness, thus potentially subjecting the event console 140 and the operator 150 to a flood of false positives. At some point, however, false positives begin to inflict their own damage by disrupting the operation of the monitored device 100 with unneeded protective measures, or by desensitizing the operator 150 to the arrival of critical alerts, i.e., true positives. Thus there is a need to improve the performance of information technology resource monitors in a way that minimizes the generation of false positives and false negatives, while preserving the capability of the monitor to unfailingly generate true positives when conditions warrant.

SUMMARY

{009} The present invention improves the performance of a monitor for information technology resources by introducing typicality filters to analyze events that have the potential to trigger the generation of event indicators such as alerts. In one embodiment of the invention, a typicality filter keeps a time-dependent history of the numbers of occurrences of an associated event, wherein time is segmented into monitoring periods. At the end of each monitoring period, the number of occurrences of the associated event

is determined, and compared with the number of occurrences of that event in a subset of the historical monitoring periods. If the number of occurrences of the event in the present monitoring period exceeds the number of occurrences of the event in a predetermined proportion of the historical monitoring periods in the specified subset, for example a majority of the members of the subset, a first action is invoked; otherwise, a second action is invoked.

BRIEF DESCRIPTION OF THE DRAWINGS

{010} FIG. 1 is used for background discussion.

{011} FIG. 2 shows a configuration that includes typicality filters according to the present invention.

{012} FIGs. 3A and 3B show exemplary history tables suitable for use by a typicality filter.

{013} FIG. 4 is a flowchart that shows operational aspects of a typicality filter.

{014} FIG. 5 shows an exemplary typicality filter.

DETAILED DESCRIPTION

{015} The present invention improves the performance of monitors for information technology resources, such as monitor 110 shown in FIG. 1, by introducing typicality

filters to analyze events and act on the results of the analysis. FIG. 2 shows a configuration that includes typicality filters according to the present invention. The configuration of FIG. 2 is essentially the same as the configuration of FIG.1, where like numbers indicate like elements, excepting that FIG. 2 shows the use of typicality filters 200 rather than persistence filters 122, thereby improving the performance of the monitor 110.

{016} For a given kind of event, an associated typicality filter keeps a time-dependent history of the numbers of occurrences of the event. The history encompasses a window of time. For example, in a preferred embodiment, the history spans five days. These days need not be consecutive, however. For example, the five days may be the last five business days, or the last five Mondays, and so forth.

{017} In the operation of a typicality filter, time is segmented into monitoring periods. In a preferred embodiment, these periods are each five minutes long. Thus an exemplary history may encompass 288 such periods per day, for five days, giving a total of 1440 historical monitoring periods. For convenience, entries of the history may be organized as shown in FIG. 3A, where a history table 300 is constructed as a matrix having N rows, where N is the number of monitoring periods per day, and M columns, where M is the number of days in the history. Let the present day be day K. The columns then represent days K-1 through K-M.

{018} FIG. 3B shows a few elements of an exemplary history table 300'. In this example, the present day is October 10, and the columns correspond, from left to right in FIG. 3B, to October 9 through October 5. The exhibited rows of the history table 300' correspond, from top to bottom, to the monitoring periods beginning at 02:00,

02:05, 02:10, 02:15, 02:20, and 02:25 Universal Time. The history tables 300 and 300' are introduced here as aids to explaining the invention rather than limitations of the invention. The history may, of course, be stored and arranged in any other convenient way as well as in the way illustrated by FIGs. 3A and 3B.

{019} In this illustrative embodiment, entries of the history table 300 are the observed numbers of occurrences of the associated event during the indicated monitoring periods on the indicated days. As a running example, suppose that the event is the arrival of an inbound packet bearing a certain origination address thought to belong to a hacker. Suppose that during the monitoring period beginning at 02:10, one such packet was detected on October 5, 8, and 9, two such packets were detected on October 7, and none on October 6. Putting these numbers in chronological order from most recent to least recent gives the row (1, 1, 2, 0, 1), which corresponds to the 02:10 row of the history table 300' shown in FIG. 3B, which table is intended for use on October 10. In this way, history tables may be constructed and updated day by day.

{020} FIG. 4 shows operational aspects of a typicality filter, with reference to the configuration of FIG. 2 and the history table 300 of FIG. 3A.

{021} At the end of the present monitoring period, the number of occurrences of the associated event is determined (step 400). This is called here the present count. The present count may correspond to the number of occurrences of the event over the course of the present monitoring period, or may aggregate counts of the event over a plurality of monitoring periods. The present count is then compared with the numbers of occurrences of that event in a subset of the monitoring periods from the history table 300 (steps 410, 420). In a preferred embodiment of the invention, the subset consists of

the same monitoring period on the five most recent days. Continuing with the running example introduced above, suppose that the present day is October 10, that the present monitoring period began at 02:10, and that the event in question occurred once during that monitoring period, giving a present count of one. This occurrence of one is then compared with the entries (1, 1, 2, 0, 1) as described above with respect to history table 300', for the five previous days.

{022} The result of the comparison is determined by whether or not the present count exceeds a predetermined proportion of the counts of occurrences of the event observed in the subset of the monitoring periods. In a preferred embodiment, the predetermined proportion is a simple majority. Thus, in the running example, the count of one on October 10 in the monitoring period that began at 02:10 is compared by majority vote with the counts for the monitoring periods begin at 02:10 on the previous five days, i.e., (1, 1, 2, 0, 1).

{023} If the present count exceeds the number of occurrences of the event for the predetermined proportion of the monitoring periods of the historical subset, the count of events is judged to be atypical, and a first action is invoked (step 430); otherwise, the count is judged to be typical, and a second action is invoked (step 440). Thus, in the running example, the count of one occurrence is judged to be typical, whereas a hypothetical count of two occurrences of the event would be judged to be atypical, despite the historical observance of two occurrences of the event on October 7 during the monitoring period beginning at 02:10.

{024} Again with reference to the running example, which concerns the arrival of packets bearing an origination address thought to belong to a hacker, the first action

(step 430), which corresponds to an atypical situation, may be to instruct the protective equipment 130 to block incoming packets that bear the origination address in question, and the second action may be to log the arrival of the packet but take no further action at that time. These actions, however, are only examples used to illustrate operational aspects of the invention; the invention encompasses a wide variety of actions that would be known to those skilled in the art of network and resource management.

{025} The invention also includes the typicality filter apparatus. An exemplary typicality filter is shown in FIG. 5. The typicality filter 500 comprises an event counter 510 for determining the present count of the number of occurrences of an event; a history table 300 as described above; and logic 520 for comparing the present count with numbers of occurrences of the event in a plurality of earlier monitoring periods selected from the history table 300, invoking a first action if the present count exceeds a predetermined proportion of the numbers of occurrences of the event in the plurality of earlier monitoring periods, and invoking a second action if the present count does not exceed the predetermined proportion of the numbers of occurrences of the event in the plurality of earlier monitoring periods.

{026} In a preferred embodiment, the typicality filter 500 may be implemented using a memory and a programmable processor operating according to stored program control instructions, for example a microprocessor.

{027} Thus the present invention improves the performance of the monitor 110 by using a simple technique that takes into account the history of events experienced by the monitored device. The foregoing description of the invention is illustrative rather than limiting, however, and the invention is limited only by the claims appended here.